

REMARKS112 Objections

Item 4 – the examiner objected to the term “self-configuring”. The original specification at page 19 states that “The term “self-configuring” used herein simply means having the power to do the configuring on his own”. In the previous amendment the specification was amended by Applicant on page 19 to clarify that when it states that “self-configuring is having the power to do the configuring on *his* own”, “*his* own” refers to the user. The examiner objected that this introduces new matter and maintained the §112 rejection. It is respectfully submitted that Applicant’s amendment does not introduce new matter and that even without the amendment, the term “self-configuring” as defined by the specification at page 19 is not indefinite.

In the English language, robots and computers are not properly assigned the pronoun “his”. Rather they are assigned the pronoun “its”. This is further supported in the specification because the same paragraph on page 19 refers to the self-configuring administrator account to be a user account.

In addition, the fact that the examiner has concluded that nowhere in the specification is it explained how an artificially intelligent process could possibly self-configure itself strongly supports the interpretation of the term “self-configuring” as referring to humans. This is similar to the basic principle of contract interpretation where terms are interpreted by the meaning that is consistent with other parts of the contract.

Generally, if a term is subject to two interpretations, the one that is compatible with the rest of the document and renders the rest of the document sensible should be

chosen over an interpretation that renders the rest of the document unfathomable. The examiner selects a creative interpretation, namely that “self-configuring” means artificial intelligence, and then argues that this interpretation is unsupported by the specification. Of course it is unsupported by the specification – that meaning of the term was never contemplated! That is why the specification never explained how artificial intelligence could configure itself.

Item 5 – The examiner objected to the fact that friendly and unfriendly lists are undefined. In the previous amendment the specification was amended at page 7 to define these terms. This clarification is supported repeatedly throughout the specification. For example, at the bottom of page 9 and continuing to page 10 the original disclosure stated:

Before that happens, however, the request is first sent by the system to the first proxy server to be checked against the friendly or unfriendly list. If the proxy server simply checked the requests for approval and allowed through those requests that are from paid users who are supposed to get approved for access to the resources, then the resource web site’s address

In addition, in the very paragraph being amended, it states that “A friendly list means a list of preferred names of entities such as URLs or subsets of URLs” and that “An unfriendly list is a list of non-preferred names of entities such as URLs or subsets of URLs.”

Finally, by implication, the specification, at page 10, when referring to a case that represents the opposite configuration result for a friendly list states that “users that paid for access to the premium highly sensitive documents (“approved users”) and who should get approved for access are listed by their *trusted host name* in the proxy server’s “unfavorite” or “unfriendly” inbound list.” In computer science language, the terms “host name” and “IP address” are used interchangeably. Thus, the case that represents the

normal configuration result is where the IP address is approved for access and is therefore in the friendly list.

Prior Art Rejection

The examiner rejected the claims based on Fuh. It is respectfully submitted that this is incorrect for the following reasons.

Fuh lacks account customization with respect to client identity and only can filter out client requests based on using one particular pre-compiled access list called “The Standard Access ACL”. Fuh uses a single “Standard Access List” for all incoming client requests that is not configured per user account (it is the same for all user accounts). See Fuh col. 11, lines 28-36; col. 10, lines 28-32; col. 10 lines 49-55 and FIG. 7A, block 706. In contrast, independent claims 1 and 5 recite the “the friendly outbound list, the unfriendly outbound list, the friendly inbound list and the unfriendly inbound lists being uniquely configurable for each user account”.

Second, Fuh’s design does not support distributed access authorization such as protecting access to resources with multiple child de-referenced resources (such as html pages that contain other references to resources) such as images, script files and objects, each located on different networks.

Third, Fuh’s system introduces security risks such as Trojan Horses and non-support of networks using NAT firewalls. Fuh’s system does not re-authenticate the user after opening the passageway and during the active sessions. Therefore, Fuh cannot support concurrent multiple user account access authorization from a single client or from multiple clients behind firewalls implementing Network Address Translation (NAT).

In addition, independent claims 1 and 5 have been amended to add the phrase

each user computer in the plurality of user computers can be configured to use the first proxy server. This language is directly supported by the specification wherein it states (see underlines portions) at pages 5-6:

The term "HTTP Browser" as used in this application refers to commercially available software applications of a user that handles the user's requests to go onto the Internet. Examples of HTTP Browser software include those sold under the name Netscape Navigator and Microsoft Explorer. HTTP Browsers are located in and may be launched from the user's computer after being installed there. In the description of the system of the present invention, we sometimes call the HTTP browser the "HTTP Client" because it is the browser, not the user, that deals with the web server and there are many browser's making requests to a particular web server. It is important to note, however, that the browser applications are only one type of HTTP client. The system of the present invention is suitable for all types of HTTP clients and is not limited to browser-based deployment.

Furthermore, on page 12 of the specification, lines 12-14 it states that "The HTTP client (the browser) should be configured to use the system's first proxy server ..." See also page 9, 3rd para. and lines 10-11, page 11, lines 3-5, page 13 lines 1-4, page 15, lines 2-3, page 20, line 8 and lines 13-16.

Furthermore, FIG. 5B states that the first proxy server is installed on every computer in the LAN and that all of them are configured to use the Proxy Server.

With the amended language the claimed invention in independent claims 1 and 5 are clearly distinguishable over the prior art since Fuh (see column 2 lines 55-60) teaches away from a targeted proxy and rather teaches a transparent proxy when it states "allowing users to use remote access via the Internet without requiring advance knowledge of the IP address of the firewall router and without restricting to a particular host". Thus Fuh's router is a transparent proxy – user computers are not configured to use it.

One of the advantages of the client being configured for the proxy is that the

claimed invention is applicable to mesh networks whereas Fuh's is not. In a mesh network, (it is noted that the whole world is a mesh network including wireless services) a client might find a different router to reach the destination and Fuh's device will fail to operate because it utilizes a transparent proxy. In the claimed invention, in contrast, the client is configured to go to the network through that proxy server so a centralized access control can be enforced.

The claimed invention also is distinguishable over Fuh because unlike the claimed invention Fuh cannot enforce a centralized access control system for distributed resources such as a single web page (HTML page) which behaves like a parent resource and rerouts (de-references) other resources across the network such as the programming objects attached on exhibit A. In the claimed invention the client always goes to the proxy. In Fuh, the client does not need to go to the router. The only way Fuh can enforce that the client goes to the router is by putting the proxy between the client network and the Internet. If the client is within the Internet and the part of the distributed resource is protected within Fuh's private network, then the client will grab the rest of the distributed resource from the Internet directly without being subject to control by Fuh's device.

Also, Fuh cannot operate in a mesh network to enforce centralized access control, as noted since a client might find a different router to reach the destination and Fuh's device will fail to operate. As noted, in the claimed invention, in contrast, the client is configured to go to the network through that proxy server so a centralized access control can be enforced.

Furthermore, the claimed invention has a further advantage over Fuh. As can be seen from Table 2 of Fuh, Fuh has a list that is nondeterministic and unstable. That is,

Fuh has approved and non-approved (denied) in the same list. The result will be different depending upon the order of how you parse through the list and aggregate the results for enforcing access rules.

To demonstrate this, consider that list aggregation can be mathematically modeled by using set operations such as set differences, unions and absolute/relative complements. The proxy in Fuh's device and in the present invention process and aggregate list elements based on actions associated with each list element. Fuh's use of two possible and opposite actions on each list element will result in at least two sets each encapsulating some or all list elements with a unique action of either approved or disapproved. The result will be a set difference on two sets in which all elements have one unique action associated with it. The present invention the list elements have one unique action associated with each element. Therefore the list will be mathematically modeled as one set encapsulating all list elements as a union of some subsets. The final result will be interpreted against a universe with only one possible action associated with it. Aggregation of purely friendly and purely friendly lists are a result of set union and absolute complement operations. Aggregation of mixed lists are the result of set differences and relative/absolute complement operations. Since set union operations are commutative and set difference operations are not commutative. An example of a union of sets (where A and B are sets) that is commutative is A plus B, since it equals B plus A. An example of a difference of sets (where A and B are sets) that is not commutative is A minus B, which does not equal B minus A. Accordingly, Fuh's lists will result in unpredictable nondeterministic behavior, in contrast to the present invention which is deterministic.

The following example, based on the kind of rules that exist in Table 2 of Fuh, demonstrates its non-deterministic nature:

Let us list all the IP addresses of the resources using wild character “#” as a place holder for all allowed range of values between 0-255. Then:

1. Have all the resources within the range of the following IP addresses be initially disapproved by default (disapproved Universe): `http://#.##.#/`
2. Then include an approved list item (set) of resources within the range of IP addresses: `http://135.##.4/`
3. Then include the denial list item (set) of resources within the range of IP addresses: `http://135.A.#.4/`
4. The resultant list is a approval set of all resources within the range of IP addresses: `http://135.##.4/` excluding resources within the range of IP addresses of `http://135.A.#.4/`

However, if we change the order as follows we get a different result:

1. Have all the resources within the range of following IP addresses are initially disapproved by default (disapproved Universe): `http://#.##.#/`
2. Then include the denial list item (set) of resources within the range of IP addresses: `http://135.A.#.4/`
3. Then include an approved list item (set) of resources within the range of IP addresses: `http://135.##.4/`

The resultant list is an approval set of all resources within the range of IP addresses:

`http://135.##.4/`

Table 2 of Fuh requires use of at least two sets (oppositely signed), as in

mathematical set theory. Combining sets is not like combining numbers. When aggregating the line instructions in Table 2 of Fuh you are combining sets using differences in sets. In combining sets using the “difference” and “complement” operation, the order matters (i.e. in mathematical jargon, the operations are not commutative). To take a simple example, if you start with a set consisting of 1, 2, 3 and you take away a set consisting of 2, 3, 4 the result is 1. In contrast, if you start with the set of 2, 3, 4 and take way the set of 1, 2, 3, the result is 4.

Consequently, if Fuh’s access lists are not customized properly, the result can be different from what was intended since it is non-deterministic. In contrast, as can be seen from claim 1 wherein states “having a friendly inbound list and/or an unfriendly inbound list only one of which is active at any given time”, the lists in the claimed invention are uniform (friendly or unfriendly) and therefore since all items in the list are same the operation being performed is always additive, i.e. a union of sets. Hence the claimed invention, in contrast to Fuh, aggregates the results for enforcing rules the same regardless of order and is therefore deterministic (i.e. always result in set union operations).

Since the independent claims 1 and 5 are distinguishable over Fuh, the dependent claims 2-4, 6, 9, 10, 14-17, 20, 21, 23-24, 32-34, 51-52, 55, 56, 59-79, and 115-118 are necessarily also distinguishable over Fuh.

Item 18 – The examiner states that Fuh discloses a first proxy server programmed to check the identity of a user ... prior to checking the identity of the requesting client It is respectfully submitted that this is not true.

The claimed invention of claim 59 is distinguishable over Fuh because the claim 1

states that “the friendly inbound list and the unfriendly inbound lists being uniquely configurable for each user account”. This is not the case in Fuh because Fuh’s inbound list is not customizable for user accounts. The reason is the claimed invention (claim 59) authenticates the user and then authenticates the client (browser) whereas Fuh authenticates the client and then authenticates the user, as can be seen from Fuh’s FIG. 7A and FIG. 7B. A client may have many users. Accordingly, when Fuh’s proxy reaches the authentication of the client it does not know the identity of the user and cannot know which user list to use since he has not authenticated the user yet. Therefore Fuh cannot use a customizable inbound list. The claimed invention (claim 59) uses user account’s inbound list to authenticate the client based on the user’s identity. Fuh cannot do this.

Furthermore, unlike the present invention as defined by claim 59, Fuh would be subject to a Trojan Horse attack because once the legitimate user logs in and the pathway is open with no further user authentication, during the period of valid user session the Trojan horse can use the opened channel without any additional user authentication and the Trojan horse can get the same resources. Under claims 59-61, 68-70, 74-76, 77-79, however, “the first proxy server is programmed to check the identity of a user who logs into the first proxy server” each time the user makes a request for a resource. The differences are also evidence from a review of Fuh FIG. 7A, 7B showing data flow diagrams.

Item 19 – The examiner contends that “the first proxy server is programmed, upon a successful authentication of the user’s credential, to use a configuration of the user’s account to check the identity of the requesting client and/or requested URL against the list or lists.” This implies that that Fuh authenticates the user prior to authenticating the

client. It is respectfully submitted that in fact, Fuh authenticates client before authenticating the user. This is seen from Fuh FIGS. 7A and 7B. In fact, Fuh cannot find the user authentication cache unless it knows the client IP address.

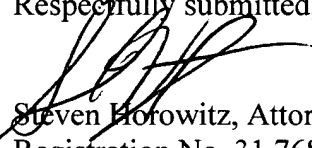
Item 22 – The examiner cites column 9 lines 20-55 and based on this the examiner claims that Fuh discloses re-routing. In fact, there is no teaching of re-routing in these lines. Fuh does not teach re-routing. The claimed invention, claims 68 and 71 do teach this. See also specification page 9, 2nd para. From top, page 10, page 18.

Since all of the foregoing amendments are understood to place the application in condition for allowance, their entry is submitted to be appropriate and is respectfully requested. It is respectfully submitted that claims 1-6, 9, 10, 14-17, 20, 21, 23-24, 32-34, 51-52, 55, 56, 59-79, and 115-118 are in condition for allowance and it is requested that they be allowed.

A check for \$930 is enclosed. This covers payment of \$525 for a response within the third month and also includes the \$405 filing fee for the RCE.

Dated: May 1, 2008

Respectfully submitted,


Steven Horowitz, Attorney for Applicant
Registration No. 31,768
295 Madison Avenue, Suite 700
New York, NY 10017
(212) 867-6800
(212) 685-6862 fax
sh@patentny.com